UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/763,422	01/26/2004	Hai Jin	1793.1189	4539

21171 7590 08/23/2007
STAAS & HALSEY LLP
SUITE 700
1201 NEW YORK AVENUE, N.W.
WASHINGTON, DC 20005

EXAMINER	
FEARER, MARK D	

ART UNIT	PAPER NUMBER
2143	

MAIL DATE	DELIVERY MODE
08/23/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/763,422	Applicant(s) JIN ET AL.	
	Examiner Mark D. Fearer	Art Unit 2143	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 26 January 2004.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-28 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-28 is/are rejected.

7)☒ Claim(s) 28 is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on 26 January 2004 is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Priority

Acknowledgment is made of applicant's claim for foreign priority under 35 U.S.C. 119(a)-(d).

Claim Objections

Claim 28 is objected to because of the following informalities: There appears to be a grammatical error. Examiner believes that 'serves' should be read as 'servers'. Also, the phrase 'the job is to created' in the abstract does not make sense. Appropriate correction is required.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was

not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 1-2, 14-15, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers").

Consider claims 1-2, 14-15, and 27. Jain et al. discloses a multi-perspective viewer for content-based interactivity (("When the system 200 is initially configured, an Environment Model (EM) process builds a skeleton static model 204 of the environment using sensor placement data. From this static model, the EM process can determine an operative range for each sensor 202 in the environment. For example, the EM process will deduce from a sensor's attributes the space in the environment that will be covered when an additional microphone is placed in the environment. During operation of the system 200, the sensor signals are received by a plurality of sensor hosts 206 associated with each sensor 202. The sensor hosts 206 comprise software servers that

recognize the source of the sensor input data. In addition, the sensor hosts 206 may include signal processing routines necessary to process the sensor input signals. Each sensor host 206 transmits the sensor information, accompanied by a sensor identifier that identifies the appropriate sensor 202, to a sensor assimilation module 208. The sensor assimilator 208 uses a sensor placement model 210 to index an input with respect to space, and, if memory permits, with respect to time.") column 9 lines 40-59) which teaches the combination of the methods of space and time with respect to movie storage. Jain et al. discloses a structure of a clip file (("As shown in FIG. 4, the preferred interactive multi-media system 300 includes a Pre-game Setup process 302, a Capture and Filtering process 304, a "Highlight Reel" Publisher process 306, and the inventive viewer process 400. In the embodiment shown in FIG. 4, the pre-game setup process comprises a utility program that is used prior to the commencement of a media event (e.g., a football game). The setup process 302 works together with the capture and filtering process 304 to automate the creation of a "highlight reel" for subsequent viewing by a user/viewer 308. In one preferred embodiment, a highlight reel is defined as a set of "important" or extraordinary plays (i.e., video clips) that are gleaned from an entire multi-media program such as an American football game. The highlight reel is "published" by the highlight reel publisher 306 and provided as input to the inventive viewer method and apparatus 400. As described in more detail below, in one preferred embodiment of the present invention, the highlight reel is published to the well-known Internet to be subsequently obtained by the viewer process. In this embodiment, the inventive viewer process 400 executes on a computer located at a user/client's home or

business. The viewer 400 causes information and multi-media information to be displayed on a user display 310 to be viewed by a user/viewer 308..") column 16 lines 1-24). Jain et al. discloses splitting video streams into clips (("FIG. 6 is a block diagram of the multi-media system of FIG. 4 showing the flow of information between the various processing blocks previously described. In the embodiment shown in FIG. 6, four video streams of information from four video cameras 316 are input into a quad splitter block 317. The quad splitter 317 creates a new composite video signal based upon the four input video streams. The composite video signal splits a video display into the four video signals at one quarter their original size. This composite video signal is provided as input to the capture process executing on a capture station (CS) 304. In addition, a Stat. Crew computer 318 provides statistical information (such as the time clock) to the CS 304 as described above. In the preferred embodiment, both the CS and the highlight reel publisher 306 comprise standard mini-tower desktop personal computers.") column 21 lines 15-30). Jain et al. discloses strategically storing video clips (("As described above, the present interactive multi-media viewing invention provides a number of innovative and useful features and functions that were heretofore unavailable to users of interactive multi-media systems. One important aspect of the inventive viewer 400 is its ability to interact with a multi-media database in an intuitive manner whereby multiple multi-media objects and events are linked together on a global timeline for subsequent accessing by the viewer. As described above with reference to the description of the capture/filter process 304 (FIG. 4), significant multi-media objects and events are preferably stored in a relational object-oriented multi-media database. The multiple data

types associated with a selected object/event are synchronized by the system 300 and thereby linked together within the database. For example, a particular video clip, audio feed, associated 3D virtual model, text and other statistical information relating to the video clip are synchronized and linked together within the multi-media database. All of the multi-media data types associated with a particular event are linked together on a global timeline. As an indexing and linking mechanism, the timeline provides a powerful, intuitive and flexible means for interacting with the multi-media system. As shown in FIG. 7, the timeline 422 is displayed in response to a user query in a format that is easily understood by the user. ") column 28 lines 50-67 and column 29 lines 1-7). However, Jain et al. fails to disclose a control file, analyzing video streams, or constructing a splitting task list. Klemets et al. discloses a control file consisting of an index and a session description protocol (("For example, some multimedia encoders capture real-time audio and video data and save the content as advanced streaming format (ASF) file (also referred to as active streaming format or advanced system format) as disclosed in U.S. Pat. No. 6,041,345. ASF is a file format specification for streaming multimedia files containing text, graphics, sound, video, and animation. An ASF file has objects including a header object containing information about the file, a data object containing the media streams (i.e., the captured audio and video data), and an optional index object that can help support random access to data within the file. The header object of an ASF file stores information as metadata that is needed by a client to decode and render the captured data. The list of streams and their relationships to each other is also stored in the header object of the ASF file. Some of the metadata items

may be mutually exclusive because the metadata items describe the same information using different spoken languages.") paragraph 0010 ("A multimedia encoder can capture real-time audio and video data and represent the captured data as multiple streams. For example, audio is typically represented as one stream and video as another. Complex files can have multiple streams, some of which may be mutually exclusive. RTSP specifies a mechanism by which a client can ask a server to deliver one or more of the encoded media streams. RTSP also provides a way for the client to obtain information about the contents of the multimedia presentation via SDP message format prior to delivery of the multimedia. SDP enumerates the available media streams and lists a limited set of auxiliary information ("SDP metadata") that is associated with the collection of streams.") paragraph 0008). Klemets et al. discloses analyzing information of streaming media source files (("In accordance with another aspect of the invention, a method streams content encoded in a streaming media format to at least one client as one or more media streams via a streaming protocol. The streaming media format has a header including one or more stream identifiers. Each of the stream identifiers corresponds to at least one of the media streams. The method includes receiving a description request from the client to describe the content. The method also includes transmitting a description message via a description protocol to the client in response to the received description request. The description message includes the header encapsulated therein. The method also includes receiving at least one of the stream identifiers from the client. The received stream identifiers correspond to the media streams selected by the client for rendering. The method further includes

delivering the selected media streams to the client via the streaming protocol in response to the received stream identifiers.") paragraph 0015). Klemets et al. discloses a program module capable of including tasks (("The invention may be described in the general context of computer-executable instructions, such as program modules, executed by one or more computers or other devices. Generally, program modules include, but are not limited to, routines, programs, objects, components, and data structures that perform particular tasks or implement particular abstract data types. The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote computer storage media including memory storage devices.") paragraph 0091). Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a system comprising a control file, is capable of analyzing video streams, and constructing a task list as taught by Klemets et al. with a system that defines the structure of a clip files, splits video streams into video clips, and strategically stores said video clips as taught by Jain et al. for the purpose of multimedia data splitting. However, Jain et al., as modified by Klemets et al., fails to disclose defining the structure of a network packet, processing the requirements of a client, defining split files, or thread creation. Halvorsen discloses defining a structure of a network packet (("From the generation of packet headers for a particular network connection, one can make some general observations. For example, in the generation of IP packets sent by a particular TCP connection, a 20 B header is added at

the front of each packet. 14 B of this header will be the same for all IP packets, and the IP length, the unique identifier, and the checksum fields (6 B in total) will probably be different for each packet. In addition, the header might contain a variable number of options.") page 46, 3.3.1). Halvorsen discloses client systems having requirements (("Traditionally, each concurrent client requires its own set of the system resources") page 4). Halvorsen discloses defining split files (("We split the headers and the media data into two files ("split-file" storage)") page 60, 4.3.6), placement strategy, and analyzing a clip file allocating requirements (("Periodic services (or enhanced pay-per-view) is a batching policy which assigns each clip a retrieval period where several clients can start at the beginning of each period to view the same movie and to share resources. Such systems are often referred to as near VoD systems.") page 30, 3.2.4) according to the split files placement strategy (("Device scheduling algorithms are different depending on hardware characteristics. For instance, a disk must take into consideration the seek time to move the head, rotational latency to position the head, and transmission time to read/write the data. These operations depend on the data placement on disk when determining the order of the disk operations to optimize efficiency. Additionally, the request deadlines are also important to avoid jitter and thereby poor playouts.") page 11, 2.5). Halvorsen discloses creating several threads to carry out successive tasks (("In many of the experiments we perform on our system, two successive runs will probably not be the same, because we measure large operations retrieving data from disk and sending it to the network. These operations generate their own kernel threads, i.e., the instructions to complete before executing our probe will

vary according to the scheduling of the threads and where in the source code the probe is placed.") pages 96-97, A2.1.1).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a system capable of defining the structure of a network packet, processing the requirements of a client, defining split files, and thread creation as taught by Halvorsen with a system comprising multimedia data splitting as taught by Jain et al., as modified by Klemets et al., for the purpose of video splitting and distributed placement.

Claims 3-4 and 16-17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") and in further view of Jin et al. ("Owl: A New Multimedia Data Splitting Scheme for Cluster Video Server").

Consider claims 3-4 and 16-17, and as applied to claims 2 and 15, respectively. Jain et al., as modified by Klemets et al. and Halvorsen, discloses a system comprising video splitting and distributed placement. However, Jain et al., as modified by Klemets et al. and Halvorsen, fails to disclose a system comprising an index file and a session description file. Jin et al. discloses a system comprising an index file and a session description protocol (SDP) file wherein the task list, the length of the video, and the name, number, ID and node source of the streaming information is defined (("SDP File: the session description protocol (SDP) [14] file. It is used for RTSP [5] describe request. The file will record the basic information of a film, such as time length, track numbers,

and track type. SDF file will be stored at the front-end machine. Fig.4 gives a sample of SDP. File Header: this structure is located at the beginning of every clip file. In this part, information about this clip file, such as track number, max bandwidth, the clip file's duration, and so on, will be recorded. Figure 5 gives the file header structure. Index File: the file describing the splitting information of a film, such as the clip numbers, clip size, every clip time scope, and the resided storage node number. It is used by front-end machine to control the distribution. This important data structure is showed in Fig.6.") pages 3-4).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a system comprising an index file and a session description file as taught by Jin et al. with a system comprising video splitting and distributed placement as taught by Jain et al., as modified by Klemets et al. and Halvorsen, for the purpose of a system comprising a distributed video storage scheme.

Claims 5 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") and in further view of Gopalakrishnan (US 6704790 B1).

Consider claims 5 and 18, and as applied to claims 1 and 14, respectively. Jain et al., as modified by Klemets et al. and Halvorsen, discloses a system comprising video splitting and distributed placement. However, Jain et al., as modified by Klemets et al. and Halvorsen, fails to disclose a system comprising a clip file structure that includes a header, an information header of media streams, and a network packet of a media

streaming service. Gopalakrishnan discloses a method wherein the structure of a clip file includes a header, an information header of media streams, and a network packet of a media streaming service ((“The header packets, such as packet 210 of stream 206 and packet 216 of stream 208, include a predetermined header designator as sent by the server 200 to indicate to the client 202 receiving the packets that a given stream is beginning and to which stream they relate. The data packets, such as packets 212a, 212b, 212c, . . . , 212n of stream 206 and packets 218a, 218b, 218c, 218d, . . . , 218n of stream 208 include a predetermined data designator as sent by the server 200 to indicate to the client 202 receiving the packets that they are data and to which stream they relate. The end-of –stream packets, such as packet 214 of stream 206 and packet 220 of stream 208 include a predetermined end-of-stream data designator as sent by the server to indicate to the client 202 receiving the packets that a given stream is ending and to which stream they relate. The server 200 also sends a switching packet 222 to indicate to the client 202 receiving the packet 222 that the server 200 is switching from a first data stream to a second data stream, namely, from stream 206 to stream 208. The switching packet 222 includes a predetermined switching designator to indicate that data streams are being switched. The switching packet is what provides embodiments of the invention the capability of server-side stream switching. The packet is sent by the server, and received by the client. Finally, those of ordinary skill within the art can appreciate that other types of packets may also be sent by the server 200 to the client 202, such as the packet 224. The packet 224 may be, for example, an HTTP message, distinguished from what is known as ASF (audio/video, or, multimedia, clip

data) packets; the invention is not so limited, however. Thus, as has been described in conjunction with FIG. 2, embodiments of the invention provide for server-side stream switching. This enables a server to indicate to a client that a first stream is going to be switched to a second stream. Such capability is typically not available in HTTP stream switching. Thus, embodiments of the invention allow for a clip of a first segment of a television program to be immediately followed by a clip of an advertisement, for example. Those of ordinary skill within the art can appreciate that other applications and advantages of the invention also exist.") column 5 lines 48-67 and column 6 lines 1-20).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a method comprising a clip file structure that includes a header, an information header of media streams, and a network packet of a media streaming service as taught by Gopalakrishnan with a system comprising video splitting and distributed placement as taught by Jain et al., as modified by Klemets et al. and Halvorsen, for the purpose of a file format for multimedia services..

Claims 6-7 and 19-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et.al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") and in further view of Kindell et al. (US 5884028 A).

Consider claims 6-7 and 19-20, and as applied to claims 1, 6, 14, and 19, respectively. Jain et al., as modified by Klemets et al. and Halvorsen, discloses a system comprising video splitting and distributed placement comprising analysis of streaming video. However, Jain et al., as modified by Klemets et al. and Halvorsen, fails

to disclose a system comprising analysis of streaming video that further comprises analyzing a number of logical units and information of the media streams thereof in the header, repeating said analysis until all logic units are accounted for, obtaining playback time, allocating storage space, or obtaining an ID of the media source files. Kindell et al. discloses a system for managing streaming data comprising obtaining control information for data transfer from a header at the beginning of a clip file, using a service thread to repeatedly read logic blocks from storage, determining a play rate, and an application program that accesses the desired data via a pointer ((“Alternatively, if the LAN resources are available, a request granted message is returned from the global resource manager and the server thread 406 begins to retrieve data from the storage device 400 in step 540 and send that data to the client thread 410 in step 542. Application program 414 can either open and begin streaming the clip from storage device 400 (reading the contents sequentially without further direct interaction between the client thread and itself) from a given location in the file or it may open the clip file, perform some random access reads and then begin streaming. Data streaming from a predetermined location is effected for those clips which contain all control information necessary for the transfer in a header at the beginning of the clip file. In this case a client thread 410 can be used to retrieve the clip information. Alternatively, some initial random access reads are used when the application program 414 must read through various locations within the file to obtain all the control information because all the control information is not located in the clip header. After the required control information is obtained the remainder of the clip is streamed. In addition to allocating

system resources, as previously mentioned, resource manager 402 controls access to the disk resource in order to help assure that the threads do not starve. This is done by requiring threads to call the resource manager and request permission before reading from the disk and to call again after the read is complete to inform the resource manager. The resource manager 402 serializes the threads and releases block read requests from the threads to the storage device 400 in sequence to insure that each thread receives equal access to the storage device 400 (as discussed above, a thread may take on a higher priority). More specifically, resource manager 402 permits a server thread 406 to obtain a block of data from a storage device 400 by presenting a block-read request generated by the server thread 406 to the operating system 407. In response to the block-read request, when the storage device 400 is free, the operating system 407 yields a semaphore (not shown) to the server thread 406 which semaphore is held by the thread 406 until the thread obtains the requested block of data from the storage device, at which time the semaphore is released. Data blocks retrieved from the storage device 400 are forwarded to the network 404 by the aforementioned network modules. Network modules are also used by the client thread 410 to read the incoming data from the network 404. The client thread then stores the clip data in the VSM buffer 416 (step 544). Server thread 406 repeatedly acquires and releases access to the storage device 400 until the entire video clip is retrieved. Data transfer in this fashion continues without intervention by the application program until the VSM buffer 416 fills to a threshold determined by the amount of data already buffered, the average rate at which the buffer is filling, and the play rate of the clip. The application program 414

begins, in step 546, the transfer of data from the buffer 416 to the video adapter (not shown) by issuing a VSMGet function call and, when the predetermined buffer threshold is reached, the application program begins playing the stream. The VSMGet function call includes as arguments the amount of data to transfer and the maximum time the application will wait to retrieve data. The call returns a pointer to the data in the buffer, and the amount of data which is currently available. When the application program 414 has transferred sufficient data in the buffer 416 to a video adapter, the application program 414 releases the portion of the buffer which was occupied by the transferred data by issuing a VSMFree function call to the buffer 416, which function call identifies the buffer section that has been released. As should be apparent, because steps 540-544 are repeated and, as just described, step 546 begins when the VSM buffer reaches a threshold, all of steps 540-546 may be executing simultaneously. It should be noted that the client thread 410 does not copy the clip data from buffer 416 into a separate memory area set aside for the application program 414. Instead, the VSM 218 provides a pointer to data (and an indication of the extent of the data) that the application program 414 can pass to the display adapter (not shown) so that the display adapter (or supporting software) can copy the clip data directly from the VSM buffer 416 to the display adapter memory. When all the video clip data has been transferred to the video adapter or when the data is no longer desired (e.g. a viewer requests a different clip before the current clip has been completely shown), the application 414 terminates the video stream by issuing a VSMClose function call. This function call permits the

computer to "clean up" any left-over data and prepare for the next video clip request.") column 17 lines 1-67 and column 18 lines 1-20).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a system for managing streaming data comprising obtaining control information for data transfer from a header at the beginning of a clip file, using a service thread to repeatedly read logic blocks from storage, determining a play rate, and an application program that accesses the desired data via a pointer as taught by Kindell et al. with a system comprising video splitting and distributed placement comprising analysis of streaming video as taught by Jain et al., as modified by Klemets et al. and Halvorsen, for the purpose of a media recording device.

Claims 8 and 21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") and in further view of Madrane (US 6573907 B1).

Consider claims 8 and 21, and as applied to claims 1 and 14, respectively. Jain et al., as modified by Klemets et al. and Halvorsen, discloses a system comprising video splitting and distributed placement comprising analysis of streaming video wherein the splitting task list is produced by analyzing the media source files. However, Jain et al., as modified by Klemets et al. and Halvorsen, fails to disclose a system comprising analysis of streaming video wherein media source files are analyzed to find a space and time deviation of each clip file and a range of a serial number of the network packet. Madrane discloses a method wherein analyzing a source file comprises comparison to a

root image derived from relative space and time information of basic interface data files and OSF file chunks comprising metadata identification corresponding to each packet ("The above-mentioned cuboid is a special case of a "root image" according to the present invention. This "root image" is derived from the video sequence and conveys information concerning both the image content of the selected sub-set of frames (called below, "basic frames") and the relative "position" of that image information in time as well as space. It is to be appreciated that the "root image" is defined by information in the interface data file. The definition specifies which video frames are "basic frames" (for example, by storing the relevant frame numbers), as well as specifying the placement positions of the basic frames relative to one another within the root image. ") column 12 lines 45-56 (" An OSF file is composed by several chunks: the metadata chunks, the structure chunks, the image chunks and the annotation chunks. Each chunk is encoded with several data packets. A packet has the following binary structure: struct _TPacket [char pSync[11]; // "SYNCHRONIZE" DWORD vOsfID; // OSF identifier char vType; // Type of chunk=CHUNK_TYPE_XXX DWORD vDataSize; // Size of data DWORD vNumPacket; // Packet number DWORD vNbPacket; // Total number of packets char pData[0]; // Packet-specific data]; typedef struct _TPacket TPacket; ") column 85 lines 28-43 (" Several OSF can be transmitted on the same communication channel. In that case, packets corresponding to different OSFs can be interleaved. The vOsfID allows the identification of each packet. It allows client applications to group received packets by OSF and rebuild the original stream.") column 85 lines 47-53).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a method wherein analyzing a source file comprises comparison to a root image derived from relative space and time information of basic interface data files and OSF file chunks comprising metadata identification corresponding to each packet as taught by Madrane with a system comprising video splitting and distributed placement comprising analysis of streaming video wherein the splitting task list is produced by analyzing the media source files as taught by Jain et al., as modified by Klemets et al. and Halvorsen, for the purpose of analysis of data flow.

Claims 9 and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") and in further view of Rehg et al. (US 6675189 B2).

Consider claims 9 and 22, and as applied to claims 2 and 15, respectively. Jain et al., as modified by Klemets et al. and Halvorsen, discloses a system comprising video splitting and distributed placement comprising analysis of streaming video wherein the splitting task list is produced by analyzing the media source files. However, Jain et al., as modified by Klemets et al. and Halvorsen, fails to disclose a method wherein the splitting of the media source file comprises reading the Index file to obtain a number of clips, and creating several threads according to the obtained number. Rehg et al. discloses a method wherein objects are divided into chunks which can be referenced by an index file. These tasks may then be implemented as threads which rely on the

operating system for scheduling of system resources ((“The splitter 301, divides an item of work into M chunks, where we no longer require that M=N. In other words, the dynamic partitioning of the data here does exactly require a one-to-one correspondence between input streams and tasks. In fact, M may vary with each item provided that M.ltoreq.M.sub.max. In the static scenario, the joiner 203 knew the number of chunks for each item, and where to find them. “) column 8 lines 6-12 (“ In applying this framework to a particular task, the application can define a chunk type, and/or supply parameterized splitter, worker, and joiner methods. In other words, if only a small number of strategies are defined, these can be stored in a table, and the chunk type can be used as an index. If the number of permitted strategies is large, than the actual methods to be applied during processing can be passed along with the chunks. “) column 8 lines 28-35 (“... tasks are implemented as threads, and the run-time system relies on the operating system to effectively schedule processor resources.”) column 3 lines 5-7).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a method wherein objects are divided into chunks which can be referenced by an index file, then implemented as threads which rely on the operating system for scheduling of system resources as taught by Rehg et al. with a system comprising video splitting and distributed placement comprising analysis of streaming video wherein the splitting task list is produced by analyzing the media source files as taught by Jain et al., as modified by Klemets et al. and Halvorsen, for the purpose of operating system support for multimedia systems.

Claims 10 and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") in further view of Rehg et al. (US 6675189 B2) and in further view of Stern (US 6591247 B2).

Consider claims 10 and 23, and as applied to claims 9 and 22, respectively. Jain et al., as modified by Klemets et al., Halvorsen, and Rehg et al., discloses a system comprising video splitting and distributed placement comprising analysis of streaming video wherein the splitting task list is produced by analyzing the media source files comprising reading an Index file to obtain a number of clips, and creating several threads according to the obtained number. However, Jain et al., as modified by Klemets et al., Halvorsen, and Rehg et al., fails to disclose a method comprising reading an Index file and obtaining a play task list including several items. Stern discloses a sequential index of the video or text element to play in the list of available videos (("`INDEX` is the specific sequential index of the video or text element to play in the list of available videos. If this field is not blank, the `SEQUENCE` field is ignored. `SEQUENCE` is either: S=Play Video or Text in Sequential Order. If there is no `INDEX` specified, the next video in the list of available videos is played. R=Play Video or Text in Random Order. If there is no `INDEX` specified, a random video is chosen from the list of available videos. ") column 20 lines 44-53 (" The Site module 500 interacts with the consumer through the attached Listening Post (LP) device. The Site module 500 performs the following tasks: Startup Communicates with the MPEG decoder using MCI commands The registry entries are retrieved and the script file is opened and

parsed into memory. The script file is then closed. The output window(s) are prepared and the MPEG decoder device is opened. The `Attract` mode is started. A separate thread is started to initiate the TCP connection to the LP. ") column 24 lines 55-67 and column 25 lines 1-2).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a sequential index of the video or text element to play in the list of available videos as taught by Stern with a system comprising video splitting and distributed placement comprising analysis of streaming video wherein the splitting task list is produced by analyzing the media source files comprising reading an Index file to obtain a number of clips, and creating several threads according to the obtained number as taught by Jain et al., as modified by Klemets et al., Halvorsen, and Rehg et al., for the purpose of audiovideo streaming.

Claims 11 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") and in further view of Dyer et al. (US 6305019 B1).

Consider claims 11 and 24, and as applied to claims 1 and 14, respectively. Jain, as modified by Klemets et al. and Halvorsen, discloses a method of video splitting and allocation for clustered video servers and distributing clip files to relevant storage server nodes, according to split files placement strategy. However, Jain, as modified by Klemets et al. and Halvorsen, fails to disclose analyzing splitting time requirements. Dyer et al. discloses a splitter that performs time-based demultiplexing of an MPEG

transport stream (("The remote video session manager 616 contains an active splitter 618, a DVM module 202 having a plurality of DVMs 203, a terminal server 620, a VME chassis 216 containing the SCM and CCM, the ETHERNET hub 622, and the fiber optic transceiver 624. The active splitter 618 contains a fiber optic receiver through which the MPEG transport stream is received from the fiber optic cable 612. The splitter 618 performs time-based demultiplexing of the MPEG transport stream to recover the 32 individual transport streams. The transport streams are then coupled as serial data through serial cable bundles to the DVM module 202. Each of the serial cable bundles support four serial connections of MPEG transport data to the DVMs 203 of the DVM module 202. This DVM module 202 operates in the same manner as the DVM module discussed above with reference to FIG. 2. As discussed above, the output of each individual DVM 203 is communicated to subscriber equipment through a cable network (shown in FIG. 1). ") column 16 lines 46-63).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a splitter that performs time-based demultiplexing of an MPEG transport stream as taught by Dyer et al. with a method of video splitting and allocation for clustered video servers and distributing clip files to relevant storage server nodes, according to split files placement strategy as taught by Jain et al., as modified by Klemets et al. and Halvorsen, for the purpose of scalable video-on-demand.

Claims 12 and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in

further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") in further view of Dyer et al. (US 6305019 B1) and in further view of Russell et al. (US 20020069420 A1).

Consider claims 12 and 25, and as applied to claims 11 and 25, respectively. Jain, as modified by Klemets et al., Halvorsen, and Dyer et al., discloses a video session manager method comprising clip placement strategy. However, Jain, as modified by Klemets et al., Halvorsen, and Dyer et al., fails to disclose a method wherein the clip placement strategy includes a data placement strategy, a hot level of a source video, and an algorithm for allocating clips to the relevant storage server nodes. Russell et al. discloses a content delivery method comprising data storage location, Least Recently Used (LRU) algorithms, and hot and cold levels to classify contents that cause content items to be stored in a defined set (("In the embodiment shown in FIG. 2, the parent servers 14 provide three centralized locations for data storage. As described in more detail below, the parent servers 14 may be operated as cache servers, for cache storage and distribution of content items to the edge servers 16. Three parent cache servers 14 located at selected locations across the region may be preferred for providing a content distribution service over the Internet within a region of about the size and population of the United States, wherein one parent cache server may be located near the west coast, another parent cache server may be located near the east coast and a further parent cache server may be located in the central portion of the country. However, in other embodiments, other suitable numbers of parent cache servers 14 may be employed and distributed in any suitable manner with respect to the systems's

service region. ") paragraph 0064 ("The edge servers 16 store copies of some of the content items (such as movies) available through the service. The edge servers 16, the parent servers 14 and/or the main server 12 determine which content items (such as movies) to store, for example, using a combination of instructions originating from the main server 12 and/or a least recently used ("LRU") algorithm. In one embodiment, the edge servers also prompt authentication of a download request by the main server. ") paragraph 0068 ("In other embodiments, the main server 12 may issue instructions that classify content items in more than two ("hot" and "cold") levels. Instructions from the main server 12 may classify content items in many different levels or classifications (such as, level 1, level 2, level 3 . . . level n), where each level or classification is associated with a different set of edge servers. Thus, for example, an instruction classifying a content item as level 1 may cause that content item to be stored in all of the edge servers, while an instruction classifying another content item as a level 2 may cause that content item to be stored in a defined set (but less than all) of the edge servers. Similarly, an instruction classifying yet another content item as a level 3 may cause that content item to be stored in another defined set (but less than all) of the edge servers, and so on. In another embodiment, instead of the main server 12 issuing a classification instruction for a content item separate from the associated content item, the classification instruction may be included with the content item.") paragraph 0079).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a content delivery method comprising data storage location, Least Recently Used (LRU) algorithms, and hot and cold levels to

classify contents that cause content items to be stored in a defined set as taught by Russell et al. with a video session manager method comprising clip placement strategy as taught by Jain et al., as modified by Klemets et al., Halvorsen, and Dyer et al., for the purpose of fault tolerant stream caching.

Claims 13 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view of Klemets et al. (US 20030236912 A1) in further view of Halvorsen ("Improving I/O Performance of Multimedia Servers") in further view of Cao (US 6782550 B1) and in further view of Sugahara (US 20030118059 A1).

Consider claims 13 and 26, and as appied to claims 1 and 14, respectively. Jain, as modified by Klemets et al. and Halvorsen, discloses a method wherein the structure of the network packet complies with a streaming media data message. However, Jain, as modified by Klemets et al. and Halvorsen, fails to disclose a method wherein the structure of a network packet complies with a streaming media data message in international real-time transmission protocol, including media type head, and serial number. Cao discloses a program guide with a current-time bar comprising media head-ends, Real-Time Transport Protocol (RTP), and device profiles comprising serial numbers (("The server 106, typically operated by, a service provider, IP media provider, broadcaster or a media deliver center, can also be referred to as media head-ends. The server 106 can provide continuous media services, such as live transmission, video-on-demand and audio-on-demand, to its subscribers.") column 5 lines 38-45 ("Examples of the protocol supported in the interface 342 may include, but not be limited to, HTTP (Hypertext Transfer Protocol), RTP (Real-Time Transport Protocol), RTSP (Real-Time

Stream Control Protocol), IP (Internet Protocol), SMTP (Simple Mail Transfer Protocol), MPEG transport, RSVP (Reservation Protocol) differential services, and H.323 (Audio/Video/Data Standard).") column 11 lines 45-51 ("The device profile area 1304 lists profile information for the selected one or more devices in the device list. As illustrated in FIG. 13A, the device profile can include information such as device ID, serial number, MAC address, IP address, switch port ID, model, status, schedule turn on date, scheduled turn off date, assigned customer ID, customer information, and device list for same customer.") column 31 lines 6-12). Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a program guide with a current-time bar comprising media head-ends, Real-Time Transport Protocol (RTP), and device profiles comprising serial numbers as taught by Cao with a method wherein the structure of the network packet complies with a streaming media data message as taught by Jain, as modified by Klemets et al. and Halvorsen, for the purpose of packet timing. However, Jain, as modified by Klemets et al., Halvorsen, and Cao, fails to disclose a method comprising a time stamp, a synchronous signal, or main media data. Sugahara discloses a method for generating information signal to be recorded comprising a time stamp detector, a video signal for synchronous reproduction, and main data that is split into audio data and video data by a demultiplexer (("Thus, the reading controller 31 implements the read-out of the desired main data from the recording medium 25. In this way, the reading controller 31 reads out desired main data, that is, desired multiplexing-resultant data, from the recording medium 25. The read-out main data are sent from the reading controller 31 to

a buffer 35. The main data are stored in the buffer 35 before being outputted therefrom to a demultiplexer 36. The demultiplexer 36 separates the main data into video data and audio data. The video data are sent from the demultiplexer 36 to a time stamp detector 43.") paragraph 0115 ("A video signal, a first audio signal, and timing information for synchronous reproduction of video and audio are multiplexed into an AV multiplexing-resultant signal.") paragraph 0176).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a method for generating information signal to be recorded comprising a time stamp detector, a video signal for synchronous reproduction, and main data that is split into audio data and video data by a demultiplexer as taught by Sugahara with a program guide with a current-time bar comprising media head-ends, Real-Time Transport Protocol (RTP), and device profiles comprising serial numbers and a method wherein the structure of the network packet complies with a streaming media data message as taught by Jain et al., as modified by Klemets et al., Halvorsen, and Cao, for the purpose of harmonized standards for exchanging program material.

Claim 28 is rejected under 35 U.S.C. 103(a) as being unpatentable over Jain et al. (US 6144375 A) in view Johnson et al. (US 7143177 B1).

Consider claim 28. Jain et al. discloses a method of splitting streaming media source files (("In the embodiment shown in FIG. 6, four video streams of information from four video cameras 316 are input into a quad splitter block 317. The quad splitter

317 creates a new composite video signal based upon the four input video streams. The composite video signal splits a video display into the four video signals at one quarter their original size.") column 21 lines 17-23). However, Jain et al. fails to disclose a method of simultaneously distributing allocating schemes setting splintered video slices to different server nodes of clustered video serves, utilizing parallel processing characteristics. Johnson et al. discloses a method of concurrent presentation comprising distributed processing (("(3.7) Allows a Presentation to be Provided in Several Languages Simultaneously: The present invention's distributed network processing architecture makes it possible to present concurrently a presentation with content provided in natural languages specific to the audience members. For example, for the same presentation performance, different audience members may have the audio portion of the presentation presented in different languages, e.g., English and Japanese. Moreover, the video content (e.g., on HTML pages) can be specified so that written text provided in the presentation can be displayed in different natural languages, depending on audience member preference.") column 6 lines 47-58).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate a method of concurrent presentation comprising distributed processing as taught by Johnson et al. with a method of splitting streaming media source files as taught by Jain et al. for the purpose of audiovisual rendering.

Conclusion

Any response to this Office Action should be faxed to (571) 273-8300 or mailed to:

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Hand-delivered responses should be brought to

Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Mark Fearer whose telephone number is (571) 270-1770. The Examiner can normally be reached on Monday-Thursday from 7:30am to 5:00pm.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, David Wiley can be reached on (571) 272-3923. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status

information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free) or 571-272-4100.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist/customer service whose telephone number is (571) 272-2600.

Mark Fearer
M.D.F./mdf
August 13, 2007

DAVID WILEY
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100